

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号  
特開2000-231550  
(P2000-231550A)

(43) 公開日 平成12年8月22日 (2000.8.22)

(51) Int.Cl. <sup>7</sup>	識別記号	F I	テマコード <sup>*</sup> (参考)
G 0 6 F 15/78	5 1 0	G 0 6 F 15/78	5 1 0 C
12/08		12/08	G
12/10		12/10	E
13/28	3 1 0	13/28	3 1 0 J

審査請求 未請求 請求項の数16 O L (全 11 頁)

(21) 出願番号 特願2000-29147(P2000-29147)  
 (22) 出願日 平成12年2月7日 (2000.2.7)  
 (31) 優先権主張番号 2 4 6 4 0 6  
 (32) 優先日 平成11年2月8日 (1999.2.8)  
 (33) 優先権主張国 米国 (U S)

(71) 出願人 000003078  
株式会社東芝  
神奈川県川崎市幸区堀川町72番地  
 (72) 発明者 小山内 建 樹  
アメリカ合衆国カリフォルニア州、サンノ  
ゼ、リンコン、サークル、1060、トーシ  
バ、アメリカ、エレクトロニック、コンポ  
ーネンツ、インコーポレーテッド内  
 (74) 代理人 100064285  
弁理士 佐藤 一雄 (外3名)

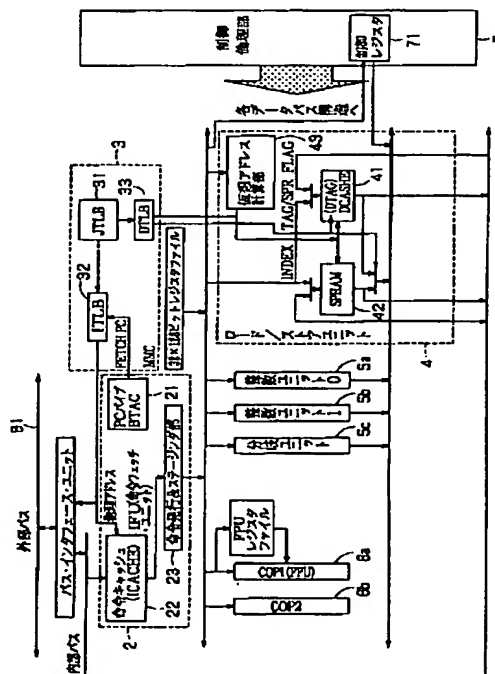
最終頁に続く

(54) 【発明の名称】 マイクロプロセッサ

## (57) 【要約】

【課題】 キャッシュメモリと同程度のレイテンシーでデータの読み書きができるRAMを備えたマイクロプロセッサの提供。

【解決手段】 本発明のマイクロプロセッサは、ロード／ストアに関連する命令を実行するLSUを備える。LSUは、外部メモリに対する読み出し／書き込みデータを一時的に格納するDCACHEと、キャッシュ以外の特殊な用途に使用されるSPRAMと、DCACHEやSPRAMをアクセスするための仮想アドレスを生成するアドレス生成器とを有する。SPRAMは、LSUのパイプラインによるロード／ストアが可能で、外部メモリとの間でDMAによるデータ転送が可能である。LSUは、DCACHEとほぼ同じレイテンシーでSPRAMにアクセスできるため、外部メモリに格納されたデータを、いったんSPRAMに格納すれば、その後、プロセッサはSPRAMにアクセスしてデータ処理を行うことができ、外部メモリよりも短時間に大量のデータを処理できる。



【特許請求の範囲】

【請求項1】ロード／ストア命令を実行するロード／ストア命令実行部を有するマイクロプロセッサにおいて、前記ロード／ストア命令実行部によるデータの読み書きが可能で、かつ、外部メモリとの間でデータをDMA(Direct Memory Access)転送可能なRAM(Random Access Memory)を備えることを特徴とするプロセッサ。

【請求項2】前記RAMは、少なくとも第1および第2のデータ格納領域を有し、前記ロード／ストア命令実行部が前記第1のデータ格納領域に対してアクセスを行っている合間に、前記第2のデータ格納領域と前記外部メモリとの間でデータのDMA転送を行い、かつ、前記ロード／ストア命令実行部が前記第2のデータ格納領域に対してアクセスを行っている合間に、前記第1のデータ格納領域と前記外部メモリとの間でデータのDMA転送を行う制御を行うRAM制御部を備えることを特徴とする請求項1に記載のマイクロプロセッサ。

【請求項3】前記RAM制御部は、前記RAMに対する前記ロード／ストア命令実行部によるアクセスとDMA転送とが競合した場合には、DMA転送を優先して行うことを特徴とする請求項2に記載のマイクロプロセッサ。

【請求項4】前記RAMに対する前記ロード／ストア命令実行部によるアクセスとDMA転送との優先度を制御可能な制御レジスタを備えることを特徴とする請求項1～3のいずれかに記載のマイクロプロセッサ。

【請求項5】前記RAMに対する前記ロード／ストア命令実行部によるアクセスとDMA転送との優先度を制御可能な外部ピンを備えることを特徴とする請求項1～3のいずれかに記載のマイクロプロセッサ。

【請求項6】前記ロード／ストア命令実行部による前記RAMのアクセスに関連するデータを、アクセス前に一時的に格納するストアバッファを備えることを特徴とする請求項1～5のいずれかに記載のマイクロプロセッサ。

【請求項7】前記RAMに対する前記ロード／ストア命令実行部によるアクセスとDMA転送とが競合した場合には、DMA転送を優先して行い、競合した前記ロード／ストア命令実行部による前記RAMのアクセスに関連するデータを一時的に前記ストアバッファに格納するRAM制御部を備えることを特徴とする請求項6に記載のマイクロプロセッサ。

【請求項8】前記RAMにアクセスするための専用の命令が設けられることを特徴とする請求項1に記載のマイクロプロセッサ。

【請求項9】前記専用の命令の中には、前記RAMに対して直接読み書きを行う命令と、外部メモリとの間でDMAによるブロック転送を行う命令との少なくとも一方が含まれることを特徴とする請求項8に記載のマイクロ

プロセッサ。

【請求項10】前記ロード／ストア命令実行部は、前記専用の命令が発行された場合には、メモリ管理ユニットの制御に先立って、該当する命令を実行することを特徴とする請求項8または9に記載のマイクロプロセッサ。

【請求項11】外部メモリから読み出したデータを一時的に格納するキャッシュメモリを備え、前記ロードストア実行部は、前記RAMと前記キャッシュメモリとを略等しいレイテンシーでアクセスすることを特徴とする請求項1～10のいずれかに記載のマイクロプロセッサ。

【請求項12】前記ロード／ストア命令実行部による前記RAMのアクセスに関連するデータを、アクセス前に一時的に格納する第1のストアバッファと、前記ロード／ストア命令実行部による前記キャッシュメモリのアクセスに関連するデータを、アクセス前に一時的に格納する第2のストアバッファと、を備えることを特徴とする請求項11に記載のマイクロプロセッサ。

【請求項13】前記RAMに格納されたデータに対応するアドレスを格納するアドレスレジスタと、前記アドレスレジスタに格納されたアドレスと、スヌープ対象となるアドレスとを比較する比較器と、を備え、前記RAMは、前記比較器による比較結果に応じた動作を行うことを特徴とする請求項12に記載のマイクロプロセッサ。

【請求項14】前記RAMは、前記比較器による比較結果が一致した場合には、一致したアドレスに対応する前記RAMのデータを、スヌープを指示したエージェントに転送することを特徴とする請求項13に記載のマイクロプロセッサ。

【請求項15】前記アドレスレジスタは、前記ロード／ストア命令実行部が前記RAMへのアクセス中であるか否かを示すフラグ情報を含んでおり、前記フラグ情報により前記ロード／ストア命令実行部が前記RAMへのアクセス中であることがわかると、スヌープを指示したエージェントに対して前記RAMへのアクセスを禁止することを特徴とする請求項14に記載のマイクロプロセッサ。

【請求項16】前記RAMがスヌープ指示信号を受領するか否かを示すフラグ情報を有し、前記RAMが前記スヌープを受領しないことを前記フラグ情報が示している場合には、前記スヌープを指示する前記エージェントのさらなるアクセスを一次的に禁止することを特徴とする請求項14または15に記載のマイクロプロセッサ。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、外部メモリよりも高速にデータの読み書きが可能なRAMを内蔵するマイクロプロセッサの内部構成に関する。

【0002】

【従来の技術】近年、プロセッサの動作速度が高くなってきたため、プロセッサと主記憶メモリ（外部メモリ）との速度差が大きくなってきた。プロセッサと主記憶メモリとの速度差を埋めるために、両者の間に小容量かつ高速のキャッシュメモリを配置することがある。最近のプロセッサの中には、この種のキャッシュメモリを内蔵したものが多い。プロセッサが要求するデータがキャッシュメモリ内に存在する場合には、キャッシュメモリから読み出したデータをプロセッサに渡すことにより、主記憶メモリへのアクセス回数が減少し、プロセッサの高速処理が可能になる。

【0003】

【発明が解決しようとする課題】しかしながら、キャッシュメモリの容量を大きくすると、キャッシュメモリ内にデータが存在するか否かを検索する時間が長くなるとともに、メモリに対するデータの読み書きに時間がかかり、メモリ・アクセスの性能が低下してしまう。このため、キャッシュメモリの容量はあまり大きくできない。したがって、容量の大きなデータの加工をキャッシュメモリ上で行おうとすると、頻繁にキャッシュリフィルを行わなければならない、プロセッサの性能が低下する。

【0004】また、キャッシュメモリは、同一のメモリアドレスに対して何度もアクセスする場合にはキャッシュ・ヒット率が向上して高速処理が可能になるが、同一のメモリアドレスに対するアクセス頻度が低い場合には、キャッシュ・ミス率が高くなり、メモリアクセスのパフォーマンスが低下する。

【0005】例えば、三次元の動画像を表示する場合には、メモリとプロセッサとの間で高速に画像データを伝送する必要があり、キャッシュメモリと同程度の速度でアクセスできるメモリに画像データを格納するのが望ましい。しかしながら、画像データは、データ容量が多く、かつ、同一のメモリアドレスに対するアクセス頻度が低い場合、キャッシュメモリに格納してもアクセス速度は上がらない。

【0006】本発明は、上記の問題点を解決するためになされたものであり、その目的は、キャッシュメモリと同程度のレイテンシーでデータの読み書きができ、かつ、キャッシュメモリとは違う目的で利用可能なRAMを備えたマイクロプロセッサを提供することにある。

【0007】

【課題を解決するための手段】上記の目的を達成するため、ロード／ストア命令を実行するロード／ストア命令実行部を有するマイクロプロセッサは、前記ロード／ストア命令実行部によるデータの読み書きが可能で、かつ、外部メモリとの間でデータをDMA (Direct Memory Access) 転送可能なRAMを備える。

【0008】請求項1の発明では、ロード／ストア実行部によるデータの読み書きが可能で、かつ、外部メモリとの間でDMAによるデータ転送を可能としたため、プ

ロセッサが画像データのような大量のデータをプログラムで処理する際の一時的な作業領域として用いることができ、その際にキャッシュメモリのような性能ペナルティがない。

【0009】請求項2の発明では、RAM内に第1および第2のデータ格納領域を設けるため、一方の格納領域にプロセッサがアクセスを行っている間に、他方の格納領域でDMA転送を行うことができる。

【0010】請求項3の発明では、ロード／ストア命令実行部によるアクセスとDMA転送とが競合した場合には、DMA転送を優先させるようにしたため、DMA転送速度が落ちるおそれがない。

【0011】請求項4の発明では、ロード／ストア命令実行部によるアクセスとDMA転送とが競合した場合の優先度を制御できるようにしたため、実装形態に応じて、最適な処理を行うことができる。

【0012】請求項5の発明では、上述した優先度を制御するための外部ピンを設けるため、外部信号により、優先度の切り替えが可能になる。

【0013】請求項6および7の発明では、上述したRAMのアクセスに関連するデータを一次的に格納するストアバッファを設けるため、RAMにアクセス可能な状態になってから、ストアバッファからデータを読み出してRAMにアクセスすることができる。

【0014】請求項8の発明では、上述したRAMにアクセスするための専用の命令を設けるため、プログラムのページ管理が簡素化される。

【0015】請求項9の発明では、RAMに対して直接読み書きする命令と、外部メモリとの間でDMAによるブロック転送を行う命令とを設けるため、請求項8の特徴に加えて、RAMと外部メモリとの間で、プログラムから明示的に高速にデータを転送できる。

【0016】請求項10の発明では、メモリ管理ユニットの制御に先立って、上述した専用の命令を実行するようにしたため、RAMアクセス命令の実行にあたってメモリ管理ユニットの動作を待つ必要がなく、RAMアクセス命令を実装する回路の高速化が可能となる。

【0017】請求項11の発明では、キャッシュメモリと略等しいレイテンシーで上述したRAMにアクセスできるため、プログラマは上述のRAMにキャッシュメモリと略等しいレイテンシーでアクセスできることを前提にプログラミングすることができる。

【0018】請求項12の発明では、上述したRAM用のストアバッファを、キャッシュメモリ用のストアバッファとは別個に設けるため、ストアバッファの制御が容易になる。

【0019】請求項13の発明では、RAMに格納されたデータと関係のあるアドレスとスヌープ対象のアドレスとを比較する比較器を設けるため、RAMに格納されたデータと関係のあるアドレスへの外部エージェントの

アクセスを容易に検出し、それに対応した動作が可能になる。

【0020】請求項14の発明では、上述した比較器で一致が検出されたRAM内の特定のデータを、スヌープを指示したエージェントに転送するようにしたため、データの検索・転送を高速に行うことができる。

【0021】請求項15の発明では、ロード/ストア命令実行部がRAMにアクセス中はスヌープ指示を受け付けないようにしたため、パイプラインストールを抑制できる。

【0022】請求項16の発明では、スヌープ処理を行うか否かを任意に設定できるようにしたため、プロセッサの動作に支障がない範囲内でスヌープ処理を行うことができる。

【0023】

【発明の実施の形態】以下、本発明に係るマイクロプロセッサについて、図面を参照しながら具体的に説明する。

【0024】(第1の実施形態)図1は本発明に係るマイクロプロセッサの一実施形態の内部構成を示すブロック図である。図1のマイクロプロセッサは、外部バスB1に接続されたバス・インタフェース・ユニット(Bus Interface Unit)1と、プロセッサが実行する命令(インストラクション)のフェッチ(取り込み)を行うIFU(Instruction Fetch Unit)2と、仮想アドレスから物理アドレスへの変換を行うMMU(Memory Management Unit)3と、ロード/ストアに関連する命令を実行するLSU(Load Store Unit)4と、ロード/ストア以外の命令を実行する複数の実行ユニット5a、5b、5cと、浮動小数点演算を行うFPU(Floating Point Unit)6a、6bと、プロセッサ各部の制御を行う制御論理部(Control Logic)7とを備える。

【0025】IFU2は、分岐命令等の命令の分岐先を記憶するBTACを参照しつつPC(Program Counter)を生成するPC-pipe21と、命令を一時的に格納するICACHE(Instruction Cache Memory)22と、命令の種類を判別してその命令を実行する実行ユニットを選択する命令発行&ステージング部(Instruction issue & Staging block)23とを有する。

【0026】MMU3は、仮想アドレスから物理アドレスへの変換を行う3つのTLB(Translation Lookaside Buffer)を有する。OS(Operating System)が有するページ・テーブルの一部のうち、物理ページ番号や保護情報など、プロセッサが必要とするアドレス変換情報がTLBに書き込まれており、これに基づいてMMU3は物理アドレスへの変換を行う。

【0027】MMU3内の3つのTLBは、JTLB(Joint Translation Lookaside Buffer)31と、ITLB(Instruction Translation Lookaside Buffer)32と、DTLB33(Data Translation Lookaside Buffer)33である。

【0028】これら3つのTLBのうち、JTLB31が本来のTLBである。JTLB31は例えば48個のエントリを有し、命令/データの別に関係なく、ページ単位で仮想アドレスをマッピングする。

【0029】一方、ITLB32とDTLB33は、総称してマイクロTLBと呼ばれる。ITLB32は命令の仮想アドレスのデータパス(Datapath)に直接接続される専用のTLBであり、DTLB33はデータの仮想アドレスのデータパスに直列接続される専用のTLBである。これらTLBは、エントリ数は少ないが、高速にアドレス変換を行う。JTLB31が生成した変換テーブルの一部が、必要に応じてITLB32またはDTLB33に転送される。

【0030】これら3つのTLBは、例えばソフトウェアによって制御される。プロセッサは、JTLB31に変換テーブルが存在しない場合に例外を起こし、例外ハンドラのプログラムがOSのページテーブルから該当するページを探し出し、JTLB31に変換テーブル100の情報を書き込む。

【0031】LSU4は、外部メモリに対する読み出し/書き込みデータを一時的に格納するDCACHE(Data Cache Memory)41と、キャッシュ以外の特殊な用途に使用されるSPRAM(Scratch Pad RAM)42と、DCACHE41やSPRAM42をアクセスするための仮想アドレスを生成するアドレス生成器(Virtual Address Computation)43とを有する。

【0032】制御論理部7は、プロセッサ各部の制御を行い、その内部には制御レジスタ(Control Register)71が設けられる。

【0033】次に、図1のマイクロプロセッサの動作を簡単に説明する。まず、PC-pipe21で生成されたPC(Program Counter)に基づいて、IFU2は命令のフェッチを行う。なお、PCは仮想アドレスである。

【0034】ITLB32は、PCを仮想アドレスから物理アドレスに変換する。物理アドレスは、ICACHE22内のITAGにおける検索に利用され、PCの指し示す命令がICACHE22内に存在するか否かを調べる。PCの指し示す命令がICACHE22内に存在しない場合(キャッシュ・ミス)のときは、物理アドレスによって不図示の外部メモリへのアクセスを行う。

【0035】具体的には、キャッシュミスすると、ICACHE22を制御するIFU2の制御論理部からBIU1の制御論理部に対して、キャッシュミスが起きたことを示す情報と、アクセスを行うべき物理アドレス情報とが渡される。BIU1は、これらの情報に基づいて外部メモリをアクセスし、アクセスが終了すると、データとキャッシュ・ラインの取得が終わった旨の信号とをIFU2に供給する。IFU2は、アクセスしたデータをICACHE22に書き込む。同時に、リフィルされたキャッシュラインのうち、PCの指し示す命令と、場合によっては、PCの指し示す命令の後に続くいくつかの命令を、命令

発行&ステージング部23に供給する。

【0036】一方、PCの指し示す命令がICACHE22内に存在する場合には、該当するICACHE22内の命令と、場合によってはPCの指し示す命令の後に続くいくつかの命令が命令発行&ステージング部23に供給される。

【0037】命令発行&ステージング部23は、命令の種類を判別し、命令を実際に実行する実行ユニット（例えば、LSU4や他の実行ユニット）を決定する。この際、命令発行&ステージング部23は、各実行ユニットの空き具合を鑑みて動的に判断する。

【0038】例えば、判別された命令がロード/ストア命令の場合には、処理できる実行ユニットはLSU4だけなので、LSU4に命令を供給できる状態になると、命令発行&ステージング部23はLSU4に命令を送る。

【0039】次に、SPRAM42に対してロード/ストアを行う場合の動作を説明する。図2は、図1のLSU4内部の一部構成を図示したものであり、SPRAM42に係る部分を制御系(Control)8とデータパス(Datapath)に分けて図示している。

【0040】図2の制御系8は、命令の流れに応じて、データパスに与える制御信号を生成する。データパスは、制御系8からの制御信号に基づいてデータを流していく。これが命令の実行に相当する。

【0041】図2において、制御系8とデータパスとをつないでいる各線は制御信号を示している。通常、制御信号は制御系8からデータパスに供給される。また、図2では省略しているが、データパスから制御系8に供給される制御信号も存在する。例えば、ICACHE22がミスしてパイプライン処理を止める場合には、データパスの一部であるICACHE22のTAGからミス信号が制御系8に供給されてパイプライン処理を停止させる。

【0042】図2において、符号10a~10jで示したブロックは、パイプライン処理を行う際にステージを区切るためのレジスタ構造を示している。これらブロックは、回路的には、フリップフロップやラッチで構成され、クロックに同期して、読み出しや書き込みを行う。以下では、これらブロックを総称してパイプラインレジスタと呼ぶ。

【0043】IFU2は、IFU制御論理部81(IFU Control)の制御を受けて、フェッチされた命令をパイプラインレジスタ10aに格納する。命令はパイプラインレジスタ10aから命令デコーダ(Instruction Decoder)82に送られる。命令デコーダ82では、命令の識別や命令の処理に必要な中間的な制御信号を生成する。中間的な制御信号を生成する理由は、命令デコーダ82でプロセッサのすべての制御信号を生成するのは量的および速度的に不可能なためであり、命令デコーダ82は、命令よりも演算器の制御信号に近く、かつ、演算器の制御信号そのものよりも抽象的な信号を生成する。

【0044】フェッチされた命令がロード/ストア命令であることが命令デコーダ82で分かった場合には、制御信号群が、関係する制御論理部に送られる。具体的には、制御信号群は、図2中のパイプライン制御論理部(Pipeline Control)83、GPR制御論理部(General Purpose Register Control)84、およびLSU制御論理部(LSU Control)85に送られる。

【0045】パイプライン制御論理部83とLSU制御論理部85では、データの流のステージに同期する形で命令や制御信号が流れていく。このため、制御系8とデータパスの双方にパイプラインレジスタ10a~10jが設けられている。

【0046】パイプライン制御論理部83は、パイプラインレジスタ10の中身の状況に応じて、パイプラインの進行の制御を行う。例えば、演算器上でリソースハザードが生じた場合には、パイプラインを停止するなどの処理を行う。

【0047】LSU制御論理部85は、ロード/ストア命令を実行するのに必要な制御信号を生成する。すなわち、GPR制御論理部84の制御によりレジスタファイル11からパイプラインレジスタ10に送られたデータは、アドレス生成器43に送られてアドレスが生成される。アドレス生成器43はアダー(Adder)制御論理部86により制御され、アドレス生成器43により生成されたアドレスは、次段のパイプラインレジスタ10に送られる。

【0048】RAM制御部に対応するSPR制御論理部(SPRAM Control)87は、パイプライン制御論理部83とLSU制御論理部85からの信号を受けて、SPRAM42へのアクセスを行う。その際、パイプラインレジスタ10中のアドレスを使用する。このアドレスがロード命令であるときは、リード・イネーブル信号を有効にし、ライト・イネーブル信号を無効にする。そして、SPRAM42からデータを読み出し、パイプラインレジスタ10に格納する。格納されたデータは、次のステージでレジスタファイル11に書き戻される。

【0049】一方、パイプラインレジスタ10中のアドレスがストア命令であるときは、リード・イネーブル信号を無効にし、ライト・イネーブル信号を有効にする。そして、レジスタファイル11からステージされてきたデータをSPRAM42に書き込む。

【0050】一方、プロセッサ外のDMA(Direct Memory Access)コントローラ9は、図示のように、外部バスを介してプロセッサ内部のBIU制御論理部(BIU Control)87に接続されている。DMAコントローラ9がBIU制御論理部88に制御信号を送ってDMA処理を開始すると、BIU制御論理部88はSPR制御論理部87に信号を送ってSPRAM42をリード/ライトさせる。すなわち、SPR制御論理部87は、BIU制御論理部88を介して間接的にDMAコントローラ9とも接続さ

れている。

【0051】SPRAM4 2に対するアクセス要求がLSU 4とBIU1 2 (BIU1 2からのアクセス要求は元はDMAコントローラ9からのものである)の双方から同時に発せられた場合は、SPR制御論理部8 7が両者を調停してイネーブル信号群を決定し、アドレス入力とデータ入力のセレクトを切り替える。

【0052】例えば、LSU 4とBIU1 2によるSPRAM4 2に対するアクセス要求の優先順位をプログラマブルに制御するには、SPR制御論理部8 7に図2に点線で示す制御レジスタ7 1を接続し、この制御レジスタ7 1の値を優先順位に応じてプログラマブルに変化させればよい。あるいは、図2に点線で示したように、プロセッサの外部ピンPに入力される信号に応じて優先順位を切り替えてもよい。

【0053】次に、SPRAM4 2の具体的な使用方法について説明する。図3はSPRAM4 2を画像データやインストラクションの加工処理に用いる例を模式的に示した図であり、以下、この図を用いてSPRAM4 2の動作を説明する。

【0054】(1)まず、図3(a)に示すように、SPRAM4 2内の約半分程度の領域に、DMA転送により、外部メモリの画像データを格納する。

【0055】(2)次に、図3(b)に示すように、SPRAM4 2に格納したデータをロード命令により汎用レジスタに読み込み、インデジャー・ユニット5.a、5.b等の実行ユニットでデータを加工処理した後、ストア命令によりSPRAM4 2に再び格納する。同時に、SPRAM4 2内の他の半分程度の領域に、DMA転送により、外部メモリの画像データを格納する。

【0056】(3)次に、図3(c)に示すように、加工終了データをDMA転送により外部メモリにストアし、同時に、SPRAM4 2内の他の半分程度の領域に格納したデータをLSU 4で加工処理し、かつ、SPRAM4 2内の半分程度の領域に外部メモリの画像データを格納する。

【0057】このように、図1のSPRAM4 2は、LSU 4のパイプラインによるロード/ストアが可能で、かつ、外部メモリとの間でDMAによるデータ転送が可能のため、画像データやインストラクションのように大量のデータを高速に処理する場合に特に適している。すなわち、SPRAM4 2は、画像データやインストラクション等の大量のデータをプロセッサで加工処理する際に、一時的なメモ・パッド領域(作業領域)として用いるのに適している。

【0058】また、LSU 4は、DCACHE 4 1とほぼ同じレイテンシー(メモリアクセスに要するシステムクロック数)でSPRAM4 2にアクセスできるため、外部メモリに格納されているデータを、いったんSPRAM4 2に格納すれば、その後、プロセッサはSPRAM4 2にアクセスしてデータ処理を行うことができ、外部メモリを直接アク

セスするよりも短時間に大量のデータを処理することが  
できる。

【0059】また、本実施形態のSPRAM4 2は、外部メモリとのデータのやり取りにDMAを用いるため、プロセッサが他の処理を行っている最中でも、外部メモリとの間でデータ転送を行える。

【0060】また、本実施形態のSPRAM4 2は、マイクロプロセッサのコア部分であるLSU 4等の各実行ユニットと同一チップ内に収容されるため、信号遅延やノイズなどの不具合が発生しにくく、また、実装面積や消費電力も低減できる。

【0061】図1では、LSU 4内にSPRAM4 2とDCACHE 4 1の双方を設ける例を説明したが、DCACHE 4 1を省略してもよい。図4はDCACHE 4 1を持たない場合のSPRAM4 2周辺の回路図である。図示のように、SPRAM4 2は、バス調停を行うBIU (Bus Interface Unit) 1 2を介してアドレス/データバスに接続され、このバスには、画像データやインストラクション等を格納する外部メモリ等が接続される。

【0062】一方、図5はSPRAM4 2とDCHACHEの双方を持つ場合のSPRAM4 2周辺の回路図である。BIU1 2には、SPRAM4 2と、DCACHE 4 1と、DCACHE 4 1用のTAG 4 4とが接続されている。DCACHE 4 1は、外部メモリから読み出されたデータの一部を格納するために用いられ、TAG 4 4は、DCACHE 4 1に格納されるデータの付加情報(アドレスなど)を格納するために用いられる。

【0063】図5に示すように、SPRAM4 2とDCACHE 4 1の双方を設けることにより、通常のメモリアクセスにはDCACHE 4 1を用い、画像データなどの大量のデータを加工処理する必要がある場合にはSPRAM4 2を用いるというように、処理によってSPRAM4 2とDCACHE 4 1を切り替えることにより、システム全体の処理性能を向上できる。

【0064】図6はプロセッサ内の各実行ユニット(図6ではCPUアクセスと表示)によるアクセスとDMA転送とが競合した場合の処理を示すタイミング図である。同時に複数のデバイスがアクセスを行えないワンプートのSPRAM4 2の場合、プロセッサ内の各実行ユニットによるアクセスとDMA転送とが競合した場合には、例えば、図1の制御レジスタ7 1によりアクセス優先順位が決められる。具体的には、プログラマは、特定の命令を用いて、制御レジスタ7 1内のビット値を優先順位に応じて切り替える。

【0065】図6はバスクロックの周波数がプロセッサのシステムクロックの半分の周波数の例を示しており、この場合、DMA転送はバスクロックに同期して行われ、その合間に各実行ユニットによるアクセスが行われる。このようなアクセスを行うことにより、各実行ユニットの処理がストールしなくなる。

【0066】このように、プロセッサ内の実行ユニット（例えば、LSU4のパイプライン）によるSPRAM42のアクセスはDMA転送の合間に行われるため、LSU4のパイプラインがDMA転送と同タイミングでSPRAM42にアクセスしようとした場合には、そのパイプラインはストール状態になる。

【0067】この場合、図7に示すように、SSB (SPRAM Store Buffer) 45を設けることにより、ストールを回避することができる。このSSB45には、パイプラインが実行するSPRAMへのストア命令が一時的に格納される。図7はDCACHE41を持たない場合のSSB45の接続関係を示す図、図8はSPRAM42とDCACHE41の双方が共通のストアバッファ46を有する例を示す図である。また、図9は、SPRAM42に専用のSSB45と、DCACHE41に専用のストアバッファ47とを設けた例を示す図である。

【0068】図7～図9のように、SSB45またはストアバッファ46、47を設けることにより、LSU4等の実行ユニットのストールを回避することができる。すなわち、SPRAM42に対する実行ユニットによるアクセスとDMA転送とが競合した場合には、実行ユニットのデータは、DMA転送が終了するまで、SSB45またはストアバッファ46、47に格納され、その間、パイプラインは他の処理を行うことができる。

【0069】なお、図8のように、SPRAM42とDCACHE41でストアバッファ46を共有する場合には、SPRAM42とDCACHE41のいずれかを選択する選択回路が必要になるため、図9よりもストアバッファ46の構成が複雑になる。

【0070】図10は、SSB45を備えたマイクロプロセッサにおいて、プロセッサ内の実行ユニットによるアクセスとDMAによるアクセスが競合した場合のタイミング図であり、実行ユニットのアクセスよりもDMA転送の方が優先度が高い例を示している。

【0071】例えば、時刻T1のときにDMA転送と実行ユニットPipe Aのアクセスが競合した場合には、DMA転送が優先して実行され、実行ユニットPipe AからのデータD1はいったんSSB45に格納される。その後、時刻T2になると、SSB45に格納されていた実行ユニットPipe Aの処理が実行されるとともに、アクセスしようとした実行ユニットPipe BのデータD2はSSB45に格納される。

【0072】その後、時刻T3になると、DMA転送が優先して実行される。この時点では、SSB45の内部にはすでに実行ユニットPipe BのデータD2が格納されているため、アクセスしようとした実行ユニットPipe Cはストール状態になる。

【0073】その後、時刻T4になると、SSB45に格納されていた実行ユニットPipe Bの処理が実行され、同時に、ストールしていた実行ユニットPipe Cのデータ

D3がSSB45に格納される。

【0074】図11はSPRAM42のライトタイミングを示すタイミング図である。図示のように、システムクロックCLKの2クロックごとにDMAライトが行われる。CPUバスにはシステムクロックCLKの2クロック分の長さのデータが供給される。

【0075】一方、図12はSPRAM42のリードタイミングを示すタイミング図である。図示のように、システムクロックCLKの2クロックごとにDMAリードが行われる。SPRAM42からは、システムクロックCLKの1クロック分の長さのデータが出力され、CPUバスにはシステムクロックCLKの2クロック分の長さのデータが出力される。

【0076】ところで、SPRAM42をアクセスする際には、通常のロード/ストア命令を用いるようにしてもよいが、SPRAM42をアクセスするための専用の命令を用意してもよい。このような専用の命令の例としては、SPRAMインデックス・ロードデータ命令、SPRAMインデックス・ストア命令、SPRAMインデックス・ストア・イミューディエートデータ命令、SPRAM42へのブロック転送命令、SPRAM42からのブロック転送命令などが考えられる。これらの命令が発行された場合には、MMU3の制御に優先して実行するようにすれば、SPRAM42への高速アクセスが可能になる。

【0077】SPRAM42用の専用の命令を設けることにより、プログラマは、SPRAM42を必要に応じて手軽に利用できるようになり、プログラム設計が容易になる。

【0078】（第2の実施形態）第1および第2の実施形態では、SPRAM42がスヌープ(snoop)機能を持たない例を説明したが、SPRAM42にスヌープ機能を持たせてもよい。ここで、スヌープとは、SPRAM42やDCACHE41が外部メモリに格納されるべき最新データを保持しているか否かを探索することをいう。スヌープが必要になる理由は、プロセッサが外部メモリのデータを読み出そうとしたときに、実際には外部メモリに該当するデータが存在せず、SPRAM42やDCACHE41が保持していることがあるためである。

【0079】図13はスヌープ機能を有するSPRAM42の周辺回路図である。図示のように、アドレスレジスタ48と、アドレスコンパレータ49とが新たに設けられる。アドレスレジスタ48はSPRAM42のTAGとして機能し、SPRAM42に格納されているデータのアドレスなどが格納される。このアドレスレジスタ48は、SPRAM42がn個のエントリーに分割されている場合には、エントリーの数分だけ設けられる。アドレスレジスタ48は、CPUバスに接続されており、その中身はCPUバスを介して読み出し可能である。

【0080】図13では省略しているが、アドレスレジスタ48に書き込みを行う手段が必要となる。この手段は、例えば、LSU4に対して専用の命令を与えて書き

587  
027



込みを行う。具体的には、プログラマが、特定のアドレスを指定することで、アドレスレジスタ48に書き込みを行う。あるいは、DMAによりSPRAM42にデータを書き込む際に、DMAのソース・アドレスを自動的にアドレスレジスタ48にセットしてもよい。

【0081】図13のアドレスコンパレータ49は、BIU12を介してアドレス/データバスと接続されており、アドレスレジスタ48にセットされているアドレスと、スヌープを要求したエージェントから送られたアドレスとを比較し、両者が一致した場合には、スヌープ・ヒットと判断する。スヌープ・ヒットのときにどのような動作を行うかは、システムの構成により異なる。

【0082】例えば、スヌープ・ヒットのときに、BIU1はCPUのバス権を取得して、SPRAM42内のヒットしたエントリーのデータをCPUバスに読み出す。あるいは、スヌープがヒットしたときに、何も処理を行わないようにしてもよい。

【0083】プロセッサを含んだシステム全体でのスヌープ・ヒット時の動作として、例えば以下の(1)、(2)が考えられる。

【0084】(1)スヌープを発行したエージェントがSPRAM42からデータを受け取る。

【0085】(2)スヌープを発行したエージェントは、SPRAM42がデータを送ることができるようになるまで待機する。

【0086】(2)の場合、アドレスレジスタ48に1ビット分のフラグを追加し、プロセッサ内の実行ユニットがSPRAM42を使用している場合にそのフラグをセットする。フラグは、BIU12に接続されており、実行ユニットからの命令でセット/リセットされる。フラグがセットされている間は、SPRAM42にスヌープ・ヒットすると、スヌープを発行したエージェントは待ち状態になる。

【0087】このように、SPRAM42にスヌープ機能を追加すれば、SPRAM42が現在どのようなデータを保持しているのかを外部から探索でき、また、必要に応じてSPRAM42に格納されているデータを読み出すことができる。したがって、プログラム設計が容易になる。

【0088】

【発明の効果】以上詳細に説明したように、本発明によれば、ロード/ストア実行部によるデータの読み書きが可能で、かつ、外部メモリとの間でDMAによるデータ転送が可能のため、画像データのような大量のデータを処理する際に一時的な作業領域として用いることができる。

【0089】また、ある命令セットを持つプロセッサをエミュレートする場合、エミュレートされる命令列を読み込んで、ネイティブの命令列に変換、加工する際の一時的な作業領域として用いることができる。ここで、RAM内で生成されるネイティブの命令セット内のコード

は、プロセッサの性能向上のためにRAM内でエディットすることができ、例えば、ジェネラル・パーパス・レジスタでのリード・アフター・ライト時のハザードを解決するために、命令列の順序を入れ替えることができる。

【0090】また、ストアバッファを設けることにより、ロード/ストア命令実行部によるアクセスとDMA転送とが競合した場合でも、パイプラインストールが起きなくなる。

【0091】また、RAMにスヌープ機能を持たせることにより、RAMに格納されているデータを必要に応じて外部に取り出すことができ、プログラム設計が容易になる。

【図面の簡単な説明】

【図1】本発明に係るマイクロプロセッサの一実施形態の内部構成を示すブロック図。

【図2】図1のLSU内部の一部構成を図示した図。

【図3】SPRAMを画像データの加工処理に用いる例を模式的に示した図。

【図4】DCACHEを持たない場合のSPRAM周辺の回路図。

【図5】SPRAMとDCACHEの双方を持つ場合のSPRAM周辺の回路図。

【図6】プロセッサ内の各実行ユニットによるアクセスとDMA転送とが競合した場合の処理を示すタイミング図。

【図7】DCACHEを持たない場合のSSBの接続関係を示す図。

【図8】SPRAMとDCACHEの双方が共通のストアバッファを有する例を示す図。

【図9】SPRAM専用のSSBとDCACHE専用のストアバッファを設けた例を示す図。

【図10】プロセッサ内の各実行ユニットによるアクセスとDMA転送とが競合した場合の処理を示すタイミング図。

【図11】SPRAMのライトタイミングを示すタイミング図。

【図12】SPRAMのリードタイミングを示すタイミング図。

【図13】スヌープ機能を有するSPRAM周辺の回路図。

【符号の説明】

- 1 BIU
- 2 IFU
- 3 MMU
- 4 LSU
- 5 a, 5 b, 5 c 実行ユニット
- 6 a, 6 b, 6 c FPU
- 7 制御論理部
- 21 PC-pipe
- 22 ICACHE
- 23 命令発行&ステージング部



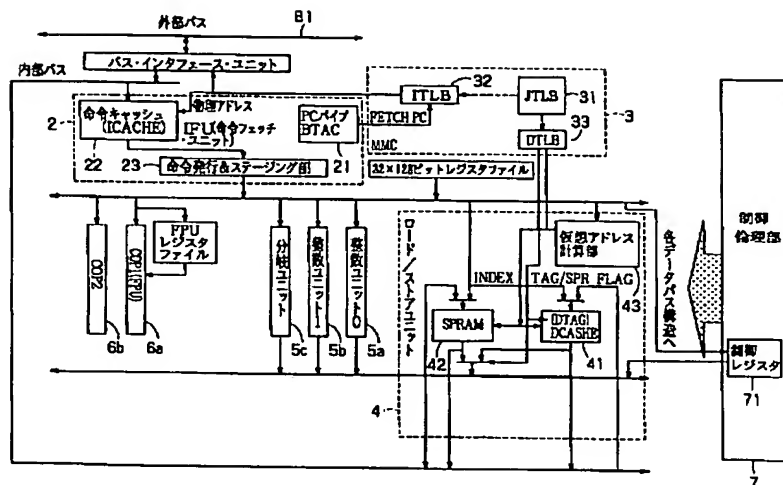
- ```

3 1 JTLB
3 2 ITLB
3 3 DTLB
4 1 データキャッシュ

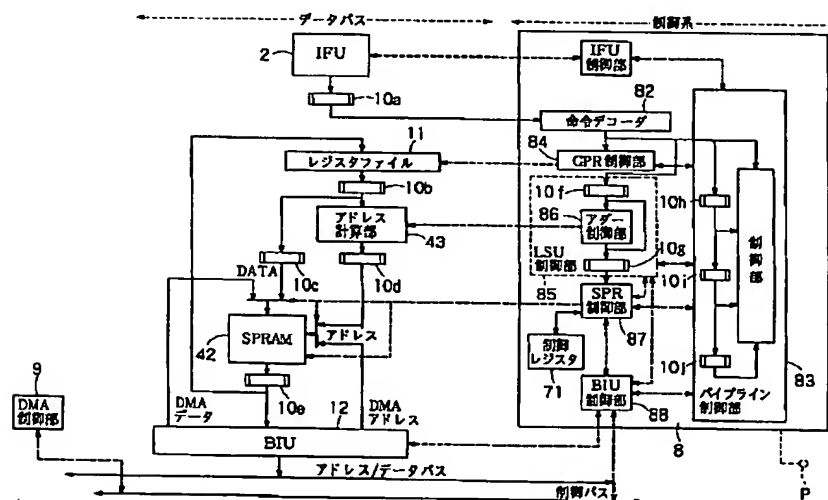
```

- 42 SPRAM  
44 データキャッシュ・タグ  
45 SSB  
46, 47 ストアバッファ

【図 1】

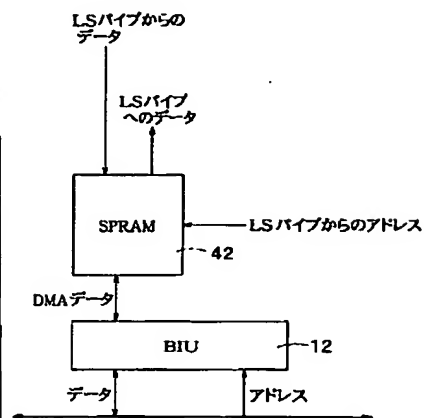


【圖2】

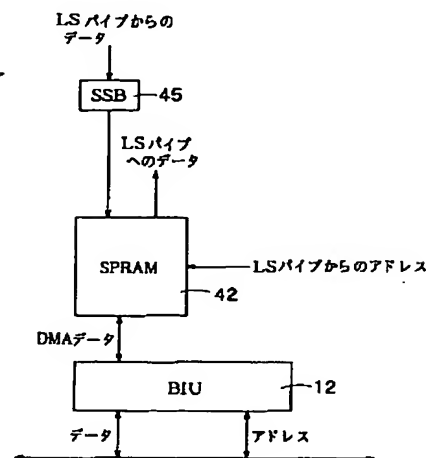


【図6】

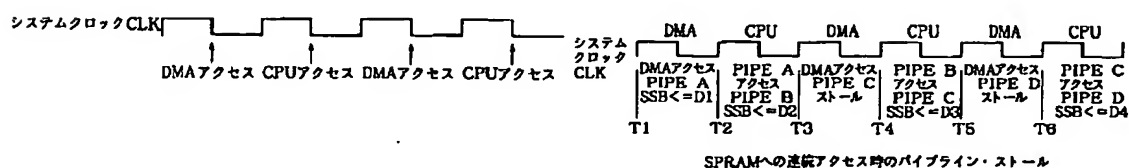
【図4】



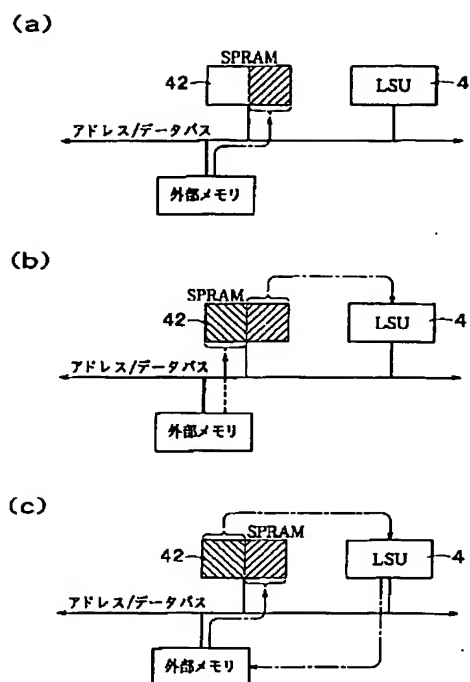
【図7】



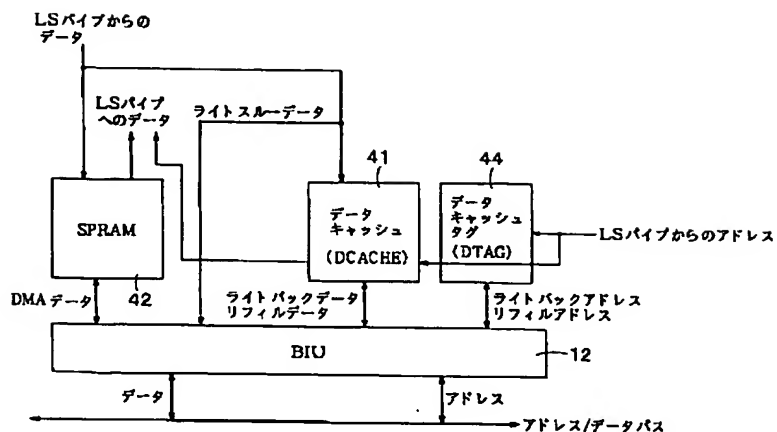
【図10】



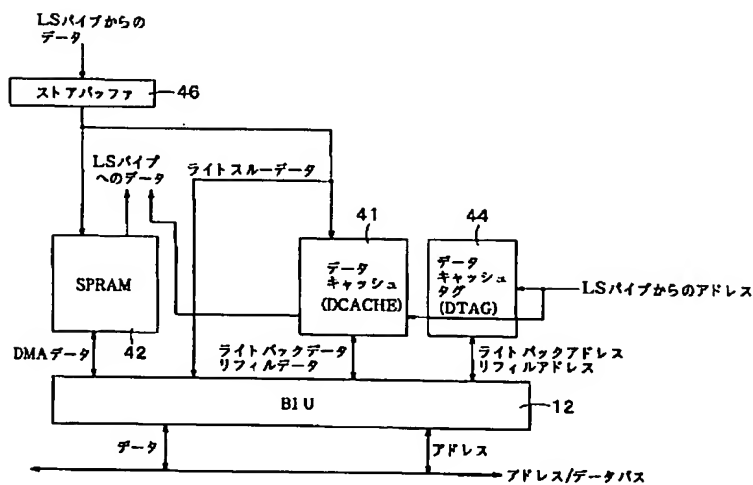
【图3】



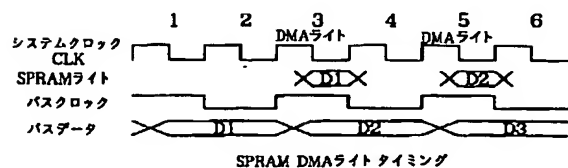
【图5】



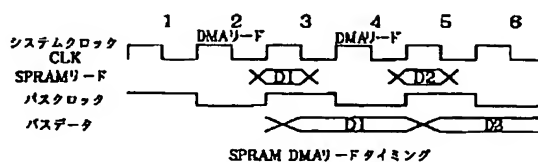
【図8】



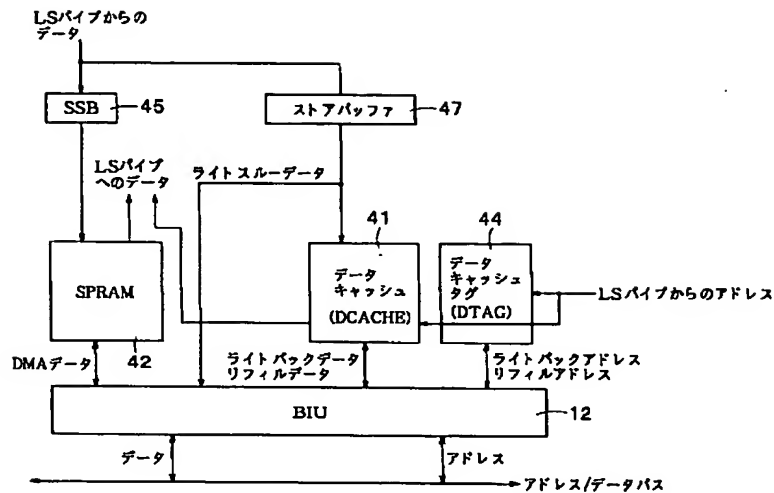
【図 1 1】



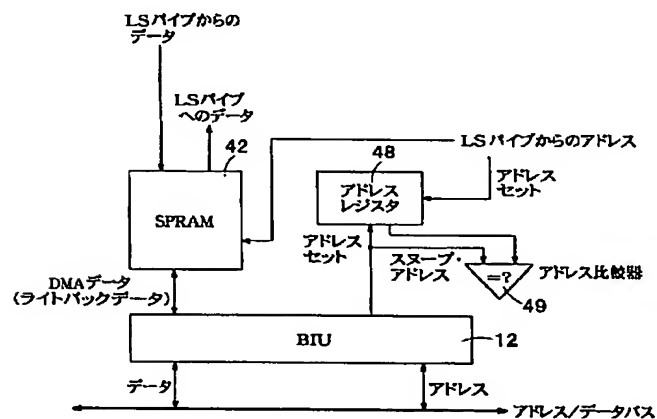
【图 12】



【図9】



【図13】



フロントページの続き

(72)発明者 内 海 透

アメリカ合衆国カリフォルニア州、サンノ  
ゼ、リンコン、サークル、1060、トーシ  
バ、アメリカ、エレクトロニック、コンポ  
ーネンツ、インコーポレーテッド内

(72)発明者 マイケル、ラーム

アメリカ合衆国カリフォルニア州、サンノ  
ゼ、リンコン、サークル、1060、トーシ  
バ、アメリカ、エレクトロニック、コンポ  
ーネンツ、インコーポレーテッド内

(72)発明者 カムラン、マリック

アメリカ合衆国カリフォルニア州、サンノ  
ゼ、リンコン、サークル、1060、トーシ  
バ、アメリカ、エレクトロニック、コンポ  
ーネンツ、インコーポレーテッド内

**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

**BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☒ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☒ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**